#include <stdio.h>

#include <conio.h>

#include <windows.h>

#include <string.h>

#include <stdlib.h>

typedef struct{

int code;

char ime[51];

char familiq[51];

int etaj;

int apartamenti;

}BLOK;

typedef struct list{

BLOK blok;

struct list \*next;

}list;

list\* addel(list\* prev)

{

list \*spisak;

int code;

int etaj;

int apartamenti;

spisak=(list \*)malloc(sizeof(list));

spisak->next=prev;

prev=spisak;

do{

printf("Въведете номер на блок: ");

fflush(stdin);

scanf("%d",&code);

if(code<0)

{

printf("Въведете положително число.\n");

continue;

}

else

{

break;

}

}while(1);

spisak->blok.code=code;

printf("Въведете име на квартал: ");

fflush(stdin);

scanf("%s",spisak->blok.ime);

printf("Въведете фамилия на управител: ");

fflush(stdin);

scanf("%s",spisak->blok.familiq);

do{

printf("Въведете етажи: ");

fflush(stdin);

scanf("%d",&etaj);

if(etaj<0)

{

printf("Въведете положително число.\n");

continue;

}

else

{

break;

}

}while(1);

spisak->blok.etaj=etaj;

do{

printf("Въведете апартаменти: ");

fflush(stdin);

scanf("%d",&apartamenti);

if(apartamenti<0)

{

printf("Въведете положително число.\n");

continue;

}

else

{

break;

}

}while(1);

spisak->blok.apartamenti=apartamenti;

return prev;

}

list\* del(list \*prev)

{

int check=0,a;

list \*head2, \*spisak;

int code;

if(prev==NULL) return prev;

do{

printf("Въведете номер на блок: ");

fflush(stdin);

scanf("%d",&code);

if(code<0)

{

printf("Въведете положително число.\n");

continue;

}

else

{

break;

}

}while(1);

for(check=0,head2=NULL,spisak=prev;spisak!=NULL;head2=spisak,spisak=spisak->next)

{

if(code==spisak->blok.code)

{

check=1;

printf("Коригирайте бройката с положително или отрицателно число: ");

scanf("%d",&a);

if((spisak->blok.etaj+a)<0) break;

if(spisak->blok.etaj+a==0)

{

if(head2!=NULL)

{

head2->next=spisak->next;

}

else

{

prev=spisak->next;

}

free(spisak);

break;

}

else

{

spisak->blok.etaj=spisak->blok.etaj+a;

break;

}

}

}

if(!check)

{

printf("\nНе беше намерен блок с такъв номер.\n\n");

}

else

{

printf("успешно изпълнена операция!");

}

return prev;

}

list\* pobroi(list\* prev)

{

list\*spisak;

BLOK \*sort,swap;

int check,broi=0,i;

char ime[51];

if(prev==NULL)return prev;

printf("Въведете име на квартал: ");

fflush(stdin);

scanf("%s",ime);

for(check=0,spisak=prev;;spisak=spisak->next)

{

if(strcmp(spisak->blok.ime,ime)==0)

{

broi++;

check=1;

}

if(spisak->next==NULL)break;

}

if(!check) printf("\n\nНе беще открит квартала!");

if(!broi) return prev;

sort=(BLOK\* )malloc(broi\*sizeof(BLOK));

for(i=0,spisak=prev;spisak!=NULL;i++,spisak=spisak->next)

{

if(strcmp(spisak->blok.ime,ime)==0)

sort[i]=spisak->blok;

}

do{

check=1;

for(i=0;i<broi;i++)

{

if(i<broi-1)

{

if(sort[i].code>sort[i+1].code)

{

swap=sort[i];

sort[i]=sort[i+1];

sort[i+1]=swap;

check=0;

}

}

}

}while(!check);

system("cls");

for(i=0;i<broi;i++)

printf("Nomer: %d\nKvartal %s\nYpravitel %s\netaji: %d\napartamenti %d\n\n",sort[i].code,sort[i].ime,sort[i].familiq ,sort[i].etaj,sort[i].apartamenti);

return prev;

}

list\* lookfor(list\* prev)

{

list \*spisak=NULL;

int check,etaj;

if(prev==NULL) return prev;

do{

printf("Въведете минимум етажи: ");

fflush(stdin);

scanf("%d",&etaj);

}while(etaj<0);

for(check=0,spisak=prev;;spisak=spisak->next)

{

if(spisak->blok.etaj>etaj)

{

printf("Nomer: %d\nKvartal %s\nYpravitel %s\netaji: %d\napartamenti %d\n\n",spisak->blok.code,spisak->blok.ime,spisak->blok.familiq ,spisak->blok.etaj,spisak->blok.apartamenti);

check=1;

}

if(spisak->next==NULL)break;

}

if(!check) printf("\n\nНе беще открит БЛОК!");

return prev;

}

int main(void)

{

list \*prev=NULL;

int ch,check=0;

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

for(;;)

{

printf ("1. Въвеждане на нов блок.\n");

printf ("2. Променяне на данни за блок.\n");

printf ("3. Извеждане данните за всички блокове в избран от потребителя квартал, по номер.\n");

printf ("4. Извеждане на блокове с етажи, повече от въведените.\n");

printf ("5. ИЗХОД \n");

printf ("Изберете опция от следното меню: ");

fflush(stdin);

ch=getche();

switch(ch)

{

case '1':

system("cls");

prev=addel(prev);

check=1;

system("Pause");

system("cls");

break;

case '2':

system("cls");

prev=del(prev);

system("Pause");

system("cls");

break;

case '3':

system("cls");

prev=pobroi(prev);

system("Pause");

system("cls");

break;

case '4':

system("cls");

prev=lookfor(prev);

system("Pause");

system("cls");

break;

case '5':

printf("\n\n");

system("Pause");

return 0;

default:

system("cls");

break;

}

}

}